

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-005791
(43)Date of publication of application : 12.01.2001

(51)Int.Cl. G06F 15/16
G06F 13/00

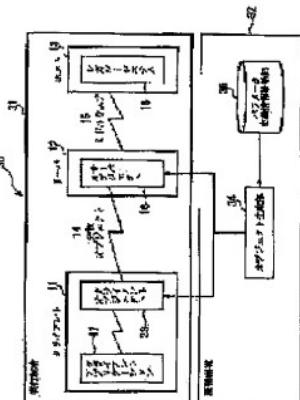
(21)Application number : 11-173375 (71)Applicant : NEC CORP
(22)Date of filing : 18.06.1999 (72)Inventor : OE SHINJI

(54) DISTRIBUTED OBJECT GENERATION SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To generate a distributed object without damaging flexibility to an application program on a client side.

SOLUTION: A client 11 is provided with a client object 33 in the execution environment 31 of a distributed computer environment 30. A client application 17 accesses a server object 16 through a distributed object 14 by using the object 33 and accesses the legacy system 18 of a host 13 by using middleware 15. A development environment 32 consists of an object generating part 34 and a parameter definition information storing part 35. The part 34 generates the objects 33 and a server object 16 on the basis of parameter definition information stored in the part 35 and outputs the objects 33 and 16 to the environment 31.



LEGAL STATUS

[Date of request for examination] 25.05.2000

[Date of sending the examiner's decision of rejection] 17.09.2002

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Data of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2001-5791

(P2001-5791A)

(43)公開日 平成13年1月12日(2001.1.12)

(51)Int.Cl. ⁷	識別記号	F I	マーク [*] (参考)
G 0 6 F 15/16 13/00	6 2 0 3 5 3	G 0 6 F 15/16 13/00	6 2 0 S 5 B 0 4 5 3 5 3 C 5 B 0 8 9

審査請求 有 検索用語の数2 O.L.(全5頁)

(21)出願番号 特願平11-173375

(71)出願人 000004237

(22)出願日 平成11年6月18日(1999.6.18)

日本電気株式会社
東京都港区芝五丁目7番1号

(72)発明者 大江 新次

東京都港区芝五丁目7番1号 日本電気株式会社内

(74)代理人 100108578

弁理士 高橋 韶男 (外3名)

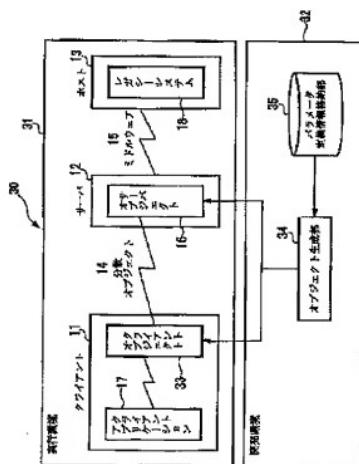
Fターム(参考) 5B045 G001
5B089 G01 GA11 CA21 GB03 GB10
JA11 KA10 KB02 KB04

(54)【発明の名称】 分散オブジェクト生成方式

(57)【要約】

【課題】 クライアント側のアプリケーションプログラムに対する汎用性を損なうことなく分散オブジェクトを生成する。

【解決手段】 分散コンピュータ環境3 0 の実行環境3 1において、クライアント1 1にクライアントオブジェクト3 3を備えた。クライアントアプリケーション1 7は、クライアントオブジェクト3 3を利用して分散オブジェクト1 4を介してサーバオブジェクト1 6を呼び出しし、ミドルウェア1 5を利用してホスト1 3のレガシーシステム1 8にアクセスする。開発環境3 2は、オブジェクト生成部3 4と、パラメータ定義情報格納部3 5とから構成した。オブジェクト生成部3 4はパラメータ定義情報格納部3 5に格納されたパラメータ定義情報に基づいてクライアントオブジェクト3 3及びサーバオブジェクト1 6を生成して実行環境3 1に出力する。



【特許請求の範囲】

【請求項1】 ミドルウェアを介してホストに結合されるサーバに、分散オブジェクトによりクライアントが結合されるなる分散コンピュータシステムにおいて、

前記クライアントは、前記ホスト上の既存システムに接続する際に使用されるクライアントアプリケーションと、前記分散オブジェクトとのインタフェースを司るクライアントオブジェクトを備えており、

前記分散オブジェクトは、前記クライアントオブジェクトと、前記サーバに備えられたサーバオブジェクトとを結合するように生成されることを特徴とする分散オブジェクト生成方式。

【請求項2】 前記クライアントは、前記クライアントオブジェクトのソースプログラムを生成し、このソースプログラムをコンパイルおよびリンクして前記クライアントオブジェクトを生成するオブジェクト生成手段を備えていることを特徴とする分散オブジェクト生成方式。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、分散コンピュータ環境において、例えばインタラネット内のシステムと、ホストコンピュータ上の既存システム(レガシーシステム)とを結合する分散オブジェクトの生成方式に関するものである。

【0002】

【従来の技術】 従来、分散コンピュータ環境におけるコネクタ機能は、インターネット内のシステムとホストコンピュータ上の既存システム(レガシーシステム)とを結合する分散オブジェクトの生成機能を有している。この分散オブジェクトを利用の方法としては以下の2つの方法が知られている。先ず、第1の方法について説明する。図8は従来技術の一例による分散コンピュータ環境1.0を示すブロック構成図である。分散コンピュータ環境1.0は、クライアント1.1と、サーバ1.2と、ホスト1.3から構成されている。ここで、クライアント1.1とサーバ1.2の間には分散オブジェクト1.4が設けられており、サーバ1.2とホスト1.3との間にはミドルウェア1.5が設けられている。サーバ1.2にはサーバオブジェクト1.6が備えられており、クライアント1.1のクライアントアプリケーション1.7は、分散オブジェクト1.4を介してサーバオブジェクト1.6を呼び出し、さらにもミドルウェア1.5を利用してホスト1.3のレガシーシステム1.8にアクセスする。

【0003】 次に、第2の方法について説明する。図4は従来技術の一例による分散コンピュータ環境2.0を示すブロック構成図である。なお、上述した従来技術と同一部分には同じ符号を配して説明を簡略または省略する。ここで、クライアント1.1には、クライアントアプリケーション1.7の前述言語に依存したラッピングモジュール1.9が設けられており、クライアントアプリケーション1.7は、ラッピングモジュール1.9を利用してホスト1.3のレガシーシステム1.8にアクセスする。

10

【0004】

【発明が解決しようとする課題】 上記の第1の方法では、クライアント1.1とサーバ1.2との間における分散オブジェクト1.4の例えば通信性能等の性能の改善および変更等を行際に、分散オブジェクト1.4のインターフェースの変更が必要となるため、分散オブジェクト1.4の手動的にクライアントアプリケーション1.7にも変更が必要となり、クライアントアプリケーション1.7とのインターフェースに影響を与えてしまうという問題がある。上記の第2の方法では、ラッピングモジュール1.9はクライアントアプリケーション1.7の記述言語に依存しており、例えば他の言語で記述されたアプリケーションプログラムから利用する場合には、記述言語間にラッピングモジュール1.9が必要となり、汎用性に欠けるという問題がある。本発明は、上記事情に鑑みてなされたもので、インターネット内のシステムと既存システムとを結合するコネクタ機能において、クライアントアプリケーションに対する分散オブジェクトの汎用性を損なうことなく分散オブジェクトを生成することが可能な分散オブジェクト生成方式を提供することを目的とする。

20

【0005】

【課題を解決するための手段】 上記課題を解決して係る目的を達成するために、請求項1に記載の本発明の分散オブジェクト生成方式は、ミドルウェアを介してホストに結合されるサーバに分散オブジェクトによりクライアントが結合されるなる分散コンピュータシステムにおいて、前記クライアントは、前記ホスト上の既存システムに接続する際に使用されるクライアントアプリケーションと、前記分散オブジェクトとのインターフェースを司るクライアントオブジェクトを備えており、前記分散オブジェクトは、前記クライアントオブジェクトと、前記サーバに備えられたサーバオブジェクトとを結合するよう生成されることを特徴としている。

30

【0006】

【0006】 さらに、請求項2に記載の分散オブジェクト生成方式は、前記クライアントは、前記クライアントオブジェクトのソースプログラムを生成し、このソースプログラムをコンパイルおよびリンクして前記クライアントオブジェクトを生成するオブジェクト生成手段を備えていることを特徴としている。

40

【0007】

【発明の実施の形態】 以下、本発明の分散オブジェクト生成方式の実施形態について添付図面を参照しながら説明する。図1は本発明の実施形態に係わる分散オブジ

50

3

エクト生成方式を具現する分散コンピュータ環境3 0 のブロック構成図であり、図2 は図1 に示すオブジェクト生成部1 2 のブロック構成図である。なお、上述した從来技術と同一部分には同じ符号を配して説明を簡略または省略する。本実施の形態による分散コンピュータ環境3 0 は、図1 に示すように、実行環境3 1 と、開発環境3 2 とから構成されている。実行環境3 1 は、例えばインターネット内のシステムから既存システムを利用するものであって、クライアント1 1 と、サーバ1 2 と、ホスト1 3 とから構成されている。

【0008】図1 に基づいてコネクタ機能を説明する。クライアント1 1 及びサーバ1 2 はインターネット内のシステムであり、ホスト1 3 はインターネット外のホストコンピュータ上のシステムであり、サーバ1 2 はホスト1 3 とミドルウェア1 5 を介して結合されている。ここで、クライアント1 1 と、サーバ1 2 とが分散オブジェクト1 4 で結合されていることによって、クライアント1 1 と、ホスト1 3 のレガシーシステム1 8 とが結合されている。クライアント1 1 はクライアントオブジェクト3 3 を備えており、クライアントアプリケーション1 7 は、クライアントオブジェクト3 3 を利用することによって、分散オブジェクト1 4 を介してサーバオブジェクト1 6 を呼び出す。そして、サーバオブジェクト1 6 はミドルウェア1 5 を利用してホスト1 3 のレガシーシステム1 8 にアクセスする。

【0009】開発環境3 2 は、オブジェクト生成部3 4 と、パラメータ定義情報格納部3 5 とから構成されており、オブジェクト生成部3 4 はパラメータ定義情報格納部3 5 に格納されたパラメータ定義情報に基づいてクライアントオブジェクト3 3 及びサーバオブジェクト1 6 を生成して実行環境3 1 に出力する。図2 に示すように、オブジェクト生成部3 4 は、サーバオブジェクトソース生成部4 1 と、クライアントオブジェクトソース生成部4 2 と、サーバオブジェクトソースコンパイルリンク部4 3 と、クライアントオブジェクトソースコンパイルリンク部4 4 とから構成されている。

【0010】本実施の形態による分散オブジェクト生成方式を具現する分散コンピュータ環境3 0 は上述の構成を備えており、次に、分散オブジェクト生成方式の動作について図1 および図2 を参照しながら説明する。まず、オブジェクト生成部3 4 は、パラメータ定義情報格納部3 2 に格納されたレガシーシステム1 0 に対するパラメータ定義情報に基づいて各オブジェクト1 6, 3 3 を生成する。サーバオブジェクトソース生成部4 1 はパラメータ定義情報に基づいてサーバオブジェクト1 6 のソースプログラムを作成し、次に、サーバオブジェクトソースコンパイルリンク部4 3 がソースプログラムからサーバオブジェクト1 6 を生成する。さらに、クライアントオブジェクトソース生成部4 2 はパラメータ定義情報に基づいてクライアントオブジェクト3 3 のソース

プログラムを作成し、次に、クライアントオブジェクトソースコンパイルリンク部4 4 がソースプログラムからクライアントオブジェクト3 3 を生成する。次に、開発環境3 2 は、作成した各オブジェクト1 6, 3 3 を実行環境3 1 に川口する。

【0011】実行環境3 1 は、開発環境3 2 から出力されたクライアントオブジェクト3 3 に基いてクライアントアプリケーション1 7 を作成し、例えばこのクライアントアプリケーション1 7 によりクライアントオブジェクト3 3 及びサーバオブジェクト1 6 を介してホスト1 3 上のレガシーシステム1 8 にアクセスする等を行うことによって、クライアントオブジェクト3 3 及びサーバオブジェクト1 6 の通信性能等に対する評価を行う。

ここで、通信性能等の性能に問題があれば、再び、サーバオブジェクトソース生成部4 1 およびクライアントオブジェクトソース生成部4 2 のそれぞれにおいて、サーバオブジェクト1 6 およびクライアントオブジェクト3 3 の各ソースプログラムを修正し、サーバオブジェクトソースコンパイルリンク部4 3においてサーバオブジェクト1 6 を、クライアントオブジェクトソースコンパイルリンク部4 4においてクライアントオブジェクト3 3 をそれぞれ生成する。なお、クライアントアプリケーション2 1 については、例えば分散オブジェクト1 4 の改善や変更等の影響が及ばないように行成する。これによって、クライアントアプリケーション2 1 の作成は一度だけで済ますことができる。また、作成した各オブジェクト1 6, 3 3 を、例えば各オブジェクト1 6, 3 3 のソースプログラムの記述言語と異なる他の記述言語によるアプリケーションプログラムから呼び出す場合、

アプリケーションプログラムの記述言語がオブジェクト対応すなわち適宜のオブジェクトを組み込んでアプリケーションプログラムを実行することができるものであれば、各オブジェクト1 6, 3 3 を修正すること無しに呼び出し可能となる。

【0012】本実施の形態による分散オブジェクト生成方式によれば、クライアント1 1 にクライアントオブジェクト3 3 が渡されており、例えばクライアント1 1 とサーバ1 2 との間の分散オブジェクト1 4 に改善や変更等を行った場合であっても、クライアントオブジェクト3 3 のみを変更することで対応することができ、クライアントアプリケーション1 7 とのインターフェースに影響を与えることを防ぐことが可能となる。さらに、クライアントオブジェクト3 3 は、オブジェクト対応している記述言語であれば呼び出し可能とされているため、クライアントアプリケーション1 7 がクライアントオブジェクト3 3 と同じ記述言語である必要はなく、クライアントアプリケーション1 7 の記述言語に対する依存度が、例えばラッピングモジュール等に比べて小さくなり、より汎用性を高めることができる。

【0013】

【発明の効果】以上説明したように、本発明の分散オブジェクト生成方式によれば、クライアントにクライアントオブジェクトが抜けられており、例えばクライアントとサーバとの間の分散オブジェクトに改善や変更等を行った場合であっても、クライアントオブジェクトのみを変更することで対応することができ、クライアントアプリケーションとのインターフェースに影響を与えることを防ぐことができる。さらに、クライアントオブジェクトは、オブジェクト対応している記述言語であれば呼び出し可能なため、クライアントアプリケーションがクライアントオブジェクトと同じ記述言語である必要はない、クライアントアプリケーションの記述言語に対する依存度が、例えはダッピングモジュール等に比べて小さくなり、より汎用性を高めることができる。

【図面の簡単な説明】

【図1】 本発明の実施形態に係わる分散オブジェクト生成方式を具現する分散コンピュータ環境のブロック構成図である。

【図2】 図1に示すオブジェクト生成部のブロック構成図である。

【図3】 従来技術の一例による分散コンピュータ環境を示すブロック構成図である。

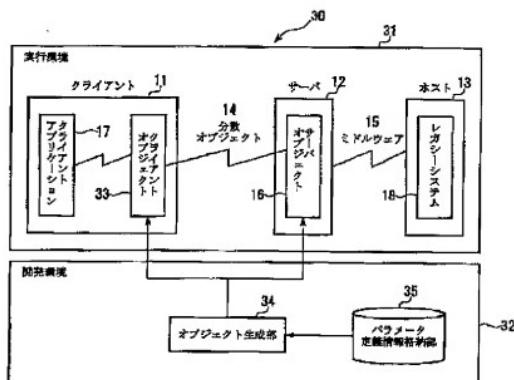
【図4】 従来技術の一例による分散コンピュータ環境を示すブロック構成図である。

【符号の説明】

- 1 1 クライアント
- 1 2 サーバ
- 1 3 ホスト
- 1 4 分散オブジェクト
- 1 5 サーバオブジェクト
- 1 6 クライアントアプリケーション
- 1 7 クライアントオブジェクト
- 1 8 クライアントオブジェクト生成部
- 1 9 パラメータ定義情報格納部
- 1 0 サーバオブジェクトソース生成部
- 1 1 クライアントオブジェクトソース生成部
- 1 2 サーバオブジェクトソースコンバイル・リンク部
- 1 3 クライアントオブジェクトソースコンバイル・リンク部

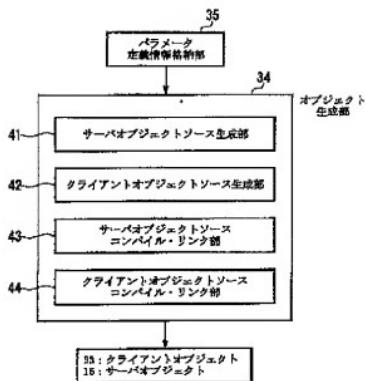
20

【図1】

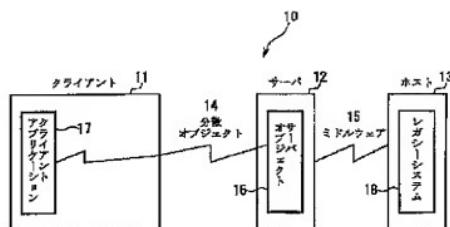


6

【 図2 】



【 図3 】



【 図4 】

